PERGAMON

Neural Networks 14 (2001) 407–425

Neural Networks

www.elsevier.com/locate/neunet

Contributed article

Learning from noisy information in FasArt and FasBack neuro-fuzzy systems

José Manuel Cano Izquierdo[a,*], Yannis A. Dimitriadis[b], Eduardo Gómez Sánchez[b], Juan López Coronado[a]

[a]*Department of Systems Engineering and Automatic Control, School of Industrial Engineering, Polytechnic University of Cartagena, Campus Muralla del Mar, 30202 Cartagena, Murcia, Spain*

[b]*Department of Signal Theory, Communications and Telematics Engineering, School of Telecommunications Engineering, University of Valladolid, Camino del Cementerio S/N, 47011 Valladolid, Spain*

Received 8 January 1999; accepted 11 January 2001

Abstract

Neuro-fuzzy systems have been in the focus of recent research as a solution to jointly exploit the main features of fuzzy logic systems and neural networks. Within the application literature, neuro-fuzzy systems can be found as methods for function identification. This approach is supported by theorems that guarantee the possibility of representing arbitrary functions by fuzzy systems. However, due to the fact that real data are often noisy, generation of accurate identifiers is presented as an important problem. Within the Adaptive Resonance Theory (ART), PROBART architecture has been proposed as a solution to this problem. After a detailed comparison of these architectures based on their design principles, the FasArt and FasBack models are proposed. They are neuro-fuzzy identifiers that offer a dual interpretation, as fuzzy logic systems or neural networks. FasArt and FasBack can be trained on noisy data without need of change in their structure or data preprocessing. In the simulation work, a comparative study is carried out on the performances of Fuzzy ARTMAP, PROBART, FasArt and FasBack, focusing on prediction error and network complexity. Results show that FasArt and FasBack clearly enhance the performance of other models in this important problem. © 2001 Elsevier Science Ltd. All rights reserved.

Keywords: Neuro-fuzzy models; System identification; Adaptive resonance theory; Learning with noise; Fuzzy ARTMAP; FasArt; FasBack

Symbols used in mathematical formulas with a description of each symbol

$\mathbf{I} = (\mathbf{a}, \mathbf{a}^c)$	Input vector, in complementary code
M	Input vector dimension
N	Number of used units in F_2 layer
$\mathbf{W}_j = \{\mathbf{w}_{ji}\}$	Weights in F_2 layer for unit j
$\mathbf{C}_j = \{\mathbf{c}_{ji}\}$	New center weights in FasArt for unit j
$\mathbf{W}_j^{ab} = \{\mathbf{w}_{ji}^{ab}\}$	Inter-ART map weights linking units j in ARTa and i in ARTb
$\mathbf{Y}$	Output of the F_2 layer
$\mathbf{X}^{ab}$	Inter-ART map activation vector
η_F	Membership function for fuzzy set F
T_j	Activation of unit j in Fuzzy ART
η_{R_j}	Activation/membership function of unit j in FasArt
η_{ji}	Activation contributed by variable i to unit j
α	Choice parameter in Fuzzy ART
β	Learning rate
β^c	Learning rate for $\mathbf{C}_j$ in FasArt
ρ	Vigilance parameter
γ	Fuzzification rate in FasArt
ϵ	Learning rate in FasBack
$\mathbf{y}$	Predicted output
$\mathbf{d}$	Desired output
$\mathbf{e}$	Error between predicted and desired output
$\mathbf{p}$	A parameter ($\mathbf{C}_j^a$, $\mathbf{C}_j^b$ or $\mathbf{W}_j^{ab}$) to be optimized in FasBack

Superscripts and subscripts a, b or ab refer to ARTa, ARTb and inter-ART, respectively.

1. Introduction

Fuzzy logic systems allow a knowledge representation close to its linguistic description, thus having transparent and easily identified performance mechanisms. However, their definition requires a complex process of knowledge extraction in order to materialize knowledge as a set of rules. On the other hand, the neural networks paradigm is appropriate for building systems with learning capabilities

* Corresponding author. Tel.: +34-968-325470; fax: +34-968-325433.
E-mail address: josem.cano@upct.es (J.M. Cano Izquierdo).

PII: S0893-6080(01)00031-4

derived from examples with numeric information. Within their effort to study the relationship and complementarities of both fuzzy logic and neural network systems, several researchers have proposed neuro-fuzzy systems that seek to combine the main characteristics of both areas. Information in fuzzy logic systems can be described in symbolic form, as rules associated with certain sets, while the information in neural network weights cannot be easily expressed in linguistic terms, so that humans may validate them by inspection. At the same time, the self-organization property of neural networks allows neuro-fuzzy systems to be built through a learning process from numeric data.

Adaptive Resonance Theory (ART) has contributed several neural architectures that can be applied to classification problems, either in a non-supervised way through ART 1 (Carpenter & Grossberg, 1989) for binary patterns and ART 2 (Carpenter & Grossberg, 1987; Carpenter, Grossberg & Rosen, 1991a) for analogue patterns, or in supervised problems through ARTMAP (Carpenter, Grossberg & Reynolds, 1991e). The introduction of concepts from fuzzy set theory, as formulated by Zadeh (1965, 1988) yielded Fuzzy ART (Carpenter, Grossberg & Rosen, 1991c,d) and Fuzzy ARTMAP models (Carpenter, Grossberg, Markuzon, Reynolds & Rosen, 1992b), that can be considered fuzzy versions of ART 1 and ARTMAP, respectively. These models have been applied to diverse problems, such as classification of handwritten characters (Carpenter, Grossberg & Iizuka, 1992a), speech (Carpenter & Govindavajan, 1993), and radar signals (Rubin, 1995), prediction of stay in hospitals (Goodman, Kaburlasos, Eubert, Carpenter & Grossberg, 1994), or rule extraction from databases (Carpenter & Tan, 1993). The properties of stability and incremental learning of both types of models were studied and proven in various research reports: in Georgiopoulos, Heileman and Huang (1992) and Georgiopoulos, Huang and Heileman (1994) for the initial ART models, and in Huang, Georgiopoulos and Heileman (1995) and Georgiopoulos, Fernlind, Bebis and Heileman (1996) for the neuro-fuzzy versions. Additionally, we should point out that there exist several hardware implementations of the basic ART modules, using parallel architectures (Malkani & Vassiliadis, 1995), optoelectronics (Blume & Esener, 1995) or VLSI systems (Serrano & Linares, 1996).

It can be easily observed from the above literature survey, that original ART models have been principally applied in classification problems. If we focus now on function identification problems, we can find a new series of models that are more or less based on the ART architectures, such as MIN–MAX (Simpson, 1993), RFALCON (Lin & Lin, 1996), PROBART (Marriott & Harrison, 1995; Srinivasa, 1997), and finally FasArt (Cano, Dimitriadis, Araúzo & Coronado, 1996a) and FasBack (Cano, Dimitriadis & Coronado, 1997) architectures studied in this paper.

The presence of noise in training data degrades the performance of ART based architectures (Williamson, 1996), causing category proliferation, especially in those which make use of the winner-take-all mechanism (Carpenter, Milenove & Noeske, 1998). Within the ART theory there are models treating the noise problem, such as dARTMAP (Carpenter et al., 1998) and PROBART (Marriott & Harrison, 1995; Srinivasa, 1997), although only the latter has been explicitly proposed as function identifier.

The rest of this paper is organized as follows: Section 2 presents a brief description of the main features of the basic Fuzzy ART and Fuzzy ARTMAP architectures, as well as the FasArt and FasBack models under study. Section 3 focuses on the problem of learning when available patterns are noisy to a certain degree. An evaluation of Fuzzy ARTMAP is made, while PROBART architecture, especially designed for this problem (Marriott & Harrison, 1995; Srinivasa, 1997) is presented and its relative merits summarized. FasArt and FasBack performances are studied under the same experimental conditions, and a comparative evaluation is presented. The final section is devoted to a discussion of the relative merits of FasArt and FasBack models for this important problem.

2. Introduction to the ART-family architectures

2.1. Basic definitions of fuzzy set theory

Fuzzy set theory introduces an extension to the classical concept of set, thus allowing treatment of vague information. This extension offers the possibility to express in a formal way linguistic assessments.

In order to clarify the contribution of FasArt and FasBack neuro-fuzzy systems, with respect to Fuzzy ARTMAP architecture, this section summarizes the basic definitions in the fuzzy set theory as introduced by Zadeh (1965) in his seminal paper, covering the formal definition of fuzzy system and the operations that can be applied to them.

Definition 1. Let U be a collection of points (objects), and u a generic element in U. Then the universe U is represented as:

$$U = \{u\}$$

Definition 2. (Fuzzy set) A fuzzy set (class) F in U is characterized by a membership function $\eta_F(u)$, $u \in U$, so that for each $u \in U$ the value $\eta_F(u) \in [0, 1]$ represent the membership degree of u to F. Formally:

$$F = \{(u, \eta_F(u)) | u \in U\}$$

This definition includes the classical definition of set (or 'crisp' set), for which a point should strictly belong or not belong to a set, i.e. $\eta_F(u) \in \{0, 1\}$.

Definition 3. (Support of a Fuzzy Set) The support of a fuzzy set F is the 'crisp' set S_F given by:

$$S_F = \{u | \eta_F(u) > 0\}$$

Definition 4. (Singleton) A fuzzy set is a singleton if its support is only one point.

Definition 5. (Complementary) The complementary of a fuzzy set F is a new fuzzy set $\bar{F}$ with membership function given by:

$$\eta_{\bar{F}} = 1 - \eta_F(u) \forall u \in U$$

The definition of union and intersection operations shown here has been proposed by Zadeh (1965), although different definitions can be found in the literature.

Definition 6. (Union) The union of two fuzzy sets F and G is a new fuzzy set $F \vee G$ whose membership function is given by:

$$\eta_{F \vee G}(u) = \max\{\eta_F(u), \eta_G(u) \forall u \in U$$

Definition 7. (Intersection) The intersection of two fuzzy sets F and g is a new fuzzy set $F \wedge G$ whose membership function is given by:

$$\eta_{F \wedge G} = \min\{\eta_F, \eta_G\} \forall u \in U$$

Definition 8. (Cartesian product) Let $F_1, \ldots, F_n$ be fuzzy sets defined respectively in $U_1, \ldots, U_n$ then the Cartesian product of $F_1, \ldots, F_n$ is a new fuzzy set defined in the product space $U_1 \times \ldots \times U_n$, with membership function:

$$\eta_{F_1 \times \cdots \times F_n} = \eta_{F_1}^{*} \cdots {}^{*} \eta_{F_n}$$

where * represents a t-norm.

Usually one of the following t-norms is selected for the implementation of the Cartesian product:

Fuzzy intersection: $u^*v = \min(u, v)$
Algebraic product: $u^*v = uv$
Drastic product: $u^*v = \max(0, u + v - 1)$

Definition 9. (Fuzzy relation) An n-ary fuzzy relation is a fuzzy set R in the product space $U_1 \times U_2 \times \ldots \times U_n$, with membership function:

$$\eta_R(u_1, \ldots, u_n)$$

where $u_i \in U_i$, $i = 1, \ldots, n$.

From this definition, we can define rules of the form:

IF u_1 IS F_1 AND … AND u_n IS F_n

where F_i is a fuzzy set defined in U_i, $i = 1, \ldots, n$ by implementing the AND by a t-norm.

Definition 10. (Composition) Given two binary fuzzy relations in U, R and S, its composition, denoted by $R\,\hat{}\,S$, is a fuzzy relation in U with membership function:

$$\eta_{R\hat{}S}(u, v) = \sup_w \{\eta_R(u, v)^* \eta_S(w, v)\}$$

Definition 11. (Composition inference rule) given a fuzzy relation R in $U \times V$, and a fuzzy set F in U, the fuzzy set induced by F in V through the fuzzy relation R is given by:

$$G = F\hat{}R$$

2.2. The basic unsupervised learning Fuzzy ART model

Among the different neural architectures that can be found in the literature, within the ART several models have been proposed with a biological inspiration reflected both in structure and functionality. Most of these models have been devoted to the problems of unsupervised clustering and supervised classification, as in the case of ART 1, ART 2, ART 3 and ARTMAP. Moreover, with the aim to introduce fuzzy logic concepts into them, Fuzzy ART and Fuzzy ARTMAP models have been proposed, which keep the main features of ART and ARTMAP, while making use of the new possibilities provided by fuzzy set theory.

The basic change of Fuzzy ART with respect to ART 1 was the substitution of the logic operators of union and intersection of crisp sets by the corresponding operators of fuzzy sets, as shown in Table 1. This change allowed Fuzzy ART to work not only with binary input patterns, but also with analogue input vectors, whose components take values in the range [0, 1].

Due to its general structure, Fuzzy ART keeps the basic characteristics of the ART family models, that make them especially attractive:

- Comparison between new inputs and already stored prototypes
- State of resonance that guarantees compliance to the stability–plasticity dilemma (Carpenter & Grossberg, 1989)
- Parallel search with significantly reduced training and performance time requirements
- Self-organization and class formation

Table 1
Comparisons between ART 1 and Fuzzy ART operators

ART 1 (binary)	Fuzzy ART (analogue)
$\cap$ = logic AND	$\wedge$ = fuzzy AND (MIN)
$\cup$ = logic OR	$\vee$ = fuzzy OR (MAX)

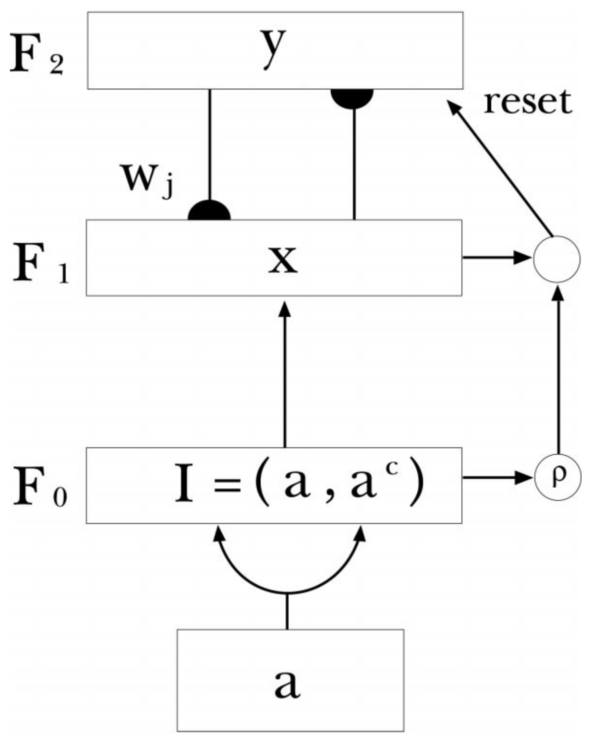

Fig. 1. Diagram of Fuzzy ART architecture. Layer F_0 performs complementary code. Layer F_1 propagates activation by bottom-up and top-down weights between F_0 and F_2, where a category choice is made using a winner-take-all mechanism.

As we can observe in Fig. 1, Fuzzy ART can be characterized as a recurrent multilevel neural network, composed of three unit levels, namely F_0, F_1 and F_2. Units in level F_0 serve as a storage site of the input pattern employing complementary code, as defined by:

$$\mathbf{I} = (\mathbf{a}, \mathbf{a}^{\mathrm{c}}) = (\mathbf{a}_1, \ldots, \mathbf{a}_{\mathrm{M}}, \mathbf{a}_1^{\mathrm{c}}, \ldots, \mathbf{a}_{\mathrm{M}}^{\mathrm{c}}) \tag{1}$$

where $\mathbf{a}_i^{\mathrm{c}} = 1 - \mathbf{a}_i$. Activity in level F_0 is transmitted to level F_1, and in the next step to level F_2 through vectors of adaptive weights $\mathbf{W}_{\mathrm{j}}$ associated with each unit j in level F_2. The activation of each F_2 unit j is calculated as:

$$T_j = \frac{|\mathbf{I} \wedge \mathbf{W}_{\mathrm{j}}|}{\alpha + |\mathbf{W}_{\mathrm{j}}|} \tag{2}$$

where α is a choice parameter.

In the competitive field of F_2, only the unit J with the highest activation, i.e. $T_J = \max\{T_j : j = 1, \ldots, N\}$ survives. If we consider that the prototype of each category j is represented by its associated weight vector $\mathbf{W}_{\mathrm{j}}$, then this mechanism can be seen as finding the category that best matches an input pattern. This process of matching evaluation fires a reset signal, if:

$$\frac{|\mathbf{I} \wedge \mathbf{W}_{\mathrm{j}}|}{|\mathbf{I}|} < \rho \tag{3}$$

where ρ is a vigilance parameter.

In the above case, matching is considered insufficient and therefore T_J takes a new value of -1. Then a new cycle begins that seeks a new winner unit. On the contrary, if:

$$\frac{|\mathbf{I} \wedge \mathbf{W}_{\mathrm{j}}|}{|\mathbf{I}|} \geq \rho \tag{4}$$

matching is sufficient and unit J modifies its associated weights, according to the following learning law:

$$\mathbf{W}_{\mathrm{J}}^{(\mathrm{new})} = \beta(\mathbf{I} \wedge \mathbf{W}_{\mathrm{J}}^{(\mathrm{old})}) + (1 - \beta)\mathbf{W}_{\mathrm{J}}^{(\mathrm{old})} \tag{5}$$

Based on the previous description of Fuzzy ART performance, we can state that it is a self-organizing, non supervised system. We should emphasize the importance of the vigilance parameter $0 \leq \rho \leq 1$, that allows us to adequately tune classification, creating coarse ($\rho \ll$) or fine categories ($\rho >$).

2.3. Supervised learning with Fuzzy ARTMAP

As already mentioned in the introductory section, ARTMAP was proposed as the first model of the ART family that allows supervised learning. Fuzzy ARTMAP, ARTMAP's fuzzy counterpart, is composed of two Fuzzy ART modules and an associative memory, called inter-ART map, as reflected in Fig. 2. Both models follow the same performance mechanisms. During the learning phase, an input pattern is presented at ARTa, while another vector (typically a label), that should be associated with the input, is presented at ARTb. During the test or prediction phase, a pattern is presented at ARTa, that through inter-ART map activates a unit in ARTb, whose weights correspond to the predicted output.

If we take a deeper look at Fuzzy ARTMAP's performance, we can see that the same categorization process occurs, when a certain pattern is presented in any of the Fuzzy ART modules. Thus, in level F_2 of Fuzzy ARTa and ARTb, we obtain two vectors, $\mathbf{Y^a}$ and $\mathbf{Y^b}$, that reflect the activation state of level F_2 units, where $y_i = 1$ if unit i is activated and 0 otherwise. Based on these vectors we can define the activation state of inter-ART map as:

$$\mathbf{X^{ab}} = \begin{cases} \mathbf{Y^b} \wedge \mathbf{W_J^{ab}} & \text{if unit } J \text{ of } F_2^a \text{ is active and } F_2^b \text{ is active} \\ \mathbf{W_J^{ab}} & \text{if node } J \text{ of } F_2^a \text{ is active and } F_2^b \text{ is not active} \\ \mathbf{Y^b} & \text{if } F_2^a \text{ is not active and } F_2^b \text{ is active} \\ 0 & \text{if neither } F_2^a \text{ nor } F_2^b \text{ are active} \end{cases} \tag{6}$$

During the learning phase, a new mechanism of inter-ART reset controls the formation of relations between categories in ARTa and ARTb, using a new inter-ART vigilance parameter ρ_{ab}. An inter-ART reset signal is fired when:

$$|\mathbf{X^{ab}}| < \rho_{ab}|\mathbf{Y^b}| \tag{7}$$

i.e. in this case we consider that the relation is not adequate, and the inter-ART reset signal deactivates the active node in level F_2 of ARTa. At the same time, the vigilance parameter

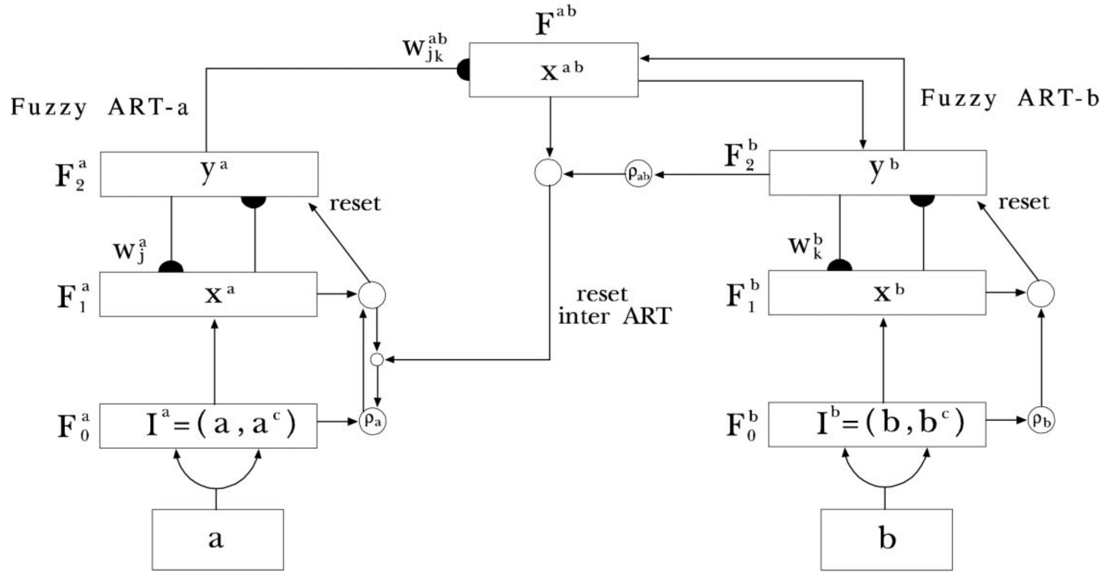

Fig. 2. Diagram of Fuzzy ARTMAP architecture. Fuzzy ARTa performs a partitioning of the input space, while Fuzzy ARTb does so on the output space. The inter-ART map, or layer F^{ab}, relates layers F_2^a and F_2^b. To ensure that correct relations are learnt, inter-ART reset may be fired during training, modifying ρ_a during one presentation.

ρ_a is increased to a new value of:

$$\rho_a = \frac{|\mathbf{I^a} \wedge \mathbf{W_J^a}|}{|\mathbf{I^a}|} \quad (8)$$

A new cycle of classification is then started in ARTa, until the criterion imposed by the inter-ART vigilance parameter ρ_{ab} is fulfilled. In that case, learning of inter-ART map weights takes place using the law of $\mathbf{W_J^{ab}} = \mathbf{X^{ab}}$.

2.4. FasArt: a new neuro-fuzzy model

The structure of Fuzzy ARTMAP, in which a map field relates categories generated in the input and output spaces by the ARTa and ARTb modules, permits its prediction performance to be seen as a rule based inference engine (Carpenter & Tan, 1993, 1995). These rules are constructed from the ARTa and ARTb category weights, and can be put in the following form (Carpenter & Tan, 1993):

IF I^a IS A_j THEN I^b IS B_k

that in turn can be decomposed into:

IF (I_1^a IS A_{j1}) AND … AND (I_n^a IS A_{jn})

THEN (I_1^b IS B_{k1}) AND … AND (I_n^b IS B_{kn})

For this interpretation to be consistent, $A_1, …, A_M$ and $B_1, …, B_M$ must be fuzzy sets in the universes where vectors I^a and I^b are defined. If these spaces are multidimensional the fuzzy sets must be decomposable, according to:

$$A_i = A_{i1} \times \cdots \times A_{in}$$

$$B_i = B_{i1} \times \cdots \times B_{in}$$

Therefore, in order to interpret Fuzzy ARTMAP as a fuzzy logic system, categories should be formally defined as fuzzy sets. This approach requires the definition of their membership functions and some expressions that describe the prediction performance as in a fuzzy logic system. This cannot be achieved with the original Fuzzy ART modules, although the Boolean logic operators in ART 1 are replaced by fuzzy logic operators, since there is not an explicit definition of the fuzzy sets on which they act.

FasArt model (Cano et al., 1996a; Sainz, Dimitriadis, Cano, Gómez & Parrado, 2000) is proposed with the aim to preserve ARTMAP general architecture, while permitting its interpretation as a fuzzy logic system. This is achieved by the introduction of an equivalence between activation function of a unit and a membership function to the fuzzy set defined by that unit. This activation-membership function η_{R_j} (T_j in ART terminology) for each unit j, is defined according to the following expression:

$$\eta_{R_j} = \prod_{i=1}^{M} \eta_{ji}(I_i) \quad (9)$$

This equation corresponds to the evaluation of the antecedent of a fuzzy rule set using the algebraic product as a t-norm to evaluate the Cartesian product (see Section 2.1). The algebraic product has been selected to implement the t-norm as in other neuro-fuzzy systems (Jang, 1993; Wang, 1994). In addition, it has the property of its derivability, which will allow in Section 2.6 to enhance learning through mathematical minimization of prediction error.

Therefore, during the prediction stage, the activation of a unit k in F_2^b is induced by rules such as

IF I IS R_j, THEN Y IS R_k

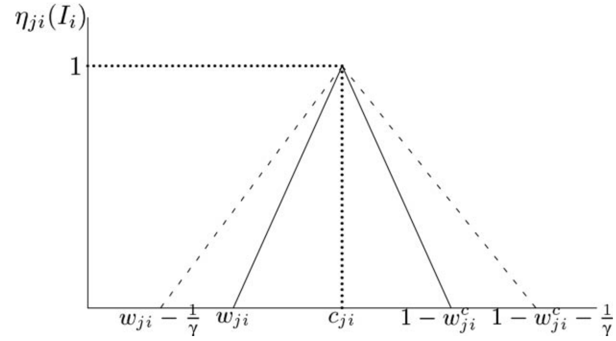

Fig. 3. The new activation/membership function of FasArt, given by Eq. (10), where w_{ji}, w_{ji}^c and c_{ji} are weights associated with unit j and feature i. Parameter γ permits increase of the fuzziness of the set, and determines the range of unknown patterns that may be learnt.

that can in turn be decomposed into:

IF (I_1 IS R_{j1}) AND … AND (I_n IS R_{jn})

THEN (y_1 IS R_{k1}) AND … AND (y_n IS R_{kn})

where vector **I** is the input vector, and R are fuzzy sets associated with units in fields F_2^a and F_2^b. Each fuzzy set R_{ji} is defined by:

$$R_{ji} = \{(I_i, \eta_{ji}(I_i)) | I_i \in U_i\}$$

where η_{ji} is given by (see Fig. 3):

$$\eta_{ij}(I_i) = \begin{cases} \max\left(0, \dfrac{\gamma(I_i - w_{ji}) + 1}{\gamma(e_{ji} - w_{ji}) + 1}\right) & \text{if } I_i \le c_{ji} \\ \max\left(0, \dfrac{\gamma(1 - I_i - w_{ji}^c) + 1}{\gamma(1 - c_{ji} - w_{ji}^c) + 1}\right) & \text{if } I_i > c_{ji} \end{cases} \tag{10}$$

where w_{ji}, w_{ji}^c and c_{ji} are weights associated with unit j in F_2.

Triangular functions have been selected for their computational simplicity, although others such as Gaussians and bell shaped could have been applied. For triangular functions, if training data represent closely the underlying data distribution, the fuzzy sets centers have also a statistical meaning and performance can be enhanced. Other approaches such as trapezoidal (or *plateau*), have been used in Abe & Lan (1995), Salzberg (1991) and Simpson (1993), but a problem may arise if a point produces maximal membership to several categories, and, thus, complex algorithms must be used to avoid class overlapping.

The new design parameter γ determines the width of the support of the fuzzy set. Then an increase in γ generates sets with a wider support, i.e. with an increased generalization capability, while a decrease in γ reduces the support and then the performance of that unit can be considered less fuzzy.

We can also observe that FasArt introduces a new weight vector, $\mathbf{C_j}$, associated with each unit j of the F_2 level. The learning law for this new set of weights is:

$$c_{ji}^{(\text{new})} = \beta^c I_i + (1 - \beta^c) c_{ji}^{(\text{old})} \tag{11}$$

where β^c is a learning rate. FasArt keeps the learning law for $\mathbf{W_j}$ from Fuzzy ART, as well as the reset, inter-ART reset and match tracking mechanisms. Therefore, FasArt preserves some of the main features of ARTMAP architectures, in particular stability and compliance to the stability–plasticity dilemma, and, thus, FasArt is suitable for incremental learning.

The interpretation of FasArt as a fuzzy logic system allows us to use a defuzzification method in order to calculate the output. In this case, we employed a defuzzification method based on the average of fuzzy sets' centers (Wang, 1994). Therefore, when an input $\mathbf{I} = (\mathbf{I}_1, \ldots, \mathbf{I}_{M^a})$ is presented to FasArt in the test phase, the output is calculated by:

$$y_m(\mathbf{I}) = \frac{\sum_{k=1}^{N^b} \sum_{j=1}^{N^b} c_{km}^b w_{jk}^{ab} \eta_{R_j}^a(\mathbf{I})}{\sum_{k=1}^{N^b} \sum_{j=1}^{N^b} w_{jk}^{ab} \eta_{R_j}^a(\mathbf{I})} \tag{12}$$

where c_{km} is the point where $\eta_{R_1}^b$ is maximum in dimension m.

In summary, it can be established that FasArt, during the prediction stage, is equivalent to a fuzzy logic system with the following specifications:

Fuzzification by single point
Inference by product
Defuzzification by average of fuzzy sets centers

Several works in the literature study the application of fuzzy logic systems to function approximation (Buckley, 1992; Castro, 1995; Castro & Delgado, 1996; Kosko, 1992). In particular, in Castro (1995) the following theorem is given:

Theorem 1. (Approximation theorem) Let $f : U \subseteq \Re \rightarrow \Re$ be a continuous function defined on compact U. If $I(a, 0) = 0$ if $a \neq 0$ (an R-implication or t-norm implication, for example), then for any $\epsilon > 0$ there exists a $S_\epsilon \in S_1$, such that $\sup\{|f(x) - S_\epsilon(x)| / x \in U\} \le \epsilon$, where S_1 is a family of fuzzy logic systems that include the systems characterized by product inferencing and single point defuzzification based on average of fuzzy sets centers.

Since FasArt is a fuzzy logic system compliant with the features required for the validity of the approximation theorem, we can conclude as a corollary, that *it is possible* for FasArt to uniformly approximate any continuous function f, defined over the compact $U, f : U \subseteq \Re^M \rightarrow V \subseteq \Re$.

These properties allowed FasArt to be applied with success in pattern recognition problems, such as handwritten recognition (Gómez, Gago, Dimitriadis, Cano & Coronado, 1998a) and document analysis (Sainz et al., 2000), and in nonlinear system identification (Cano, Dimitriadis, Araúzo, Abajo & Coronado, 1996c; Cano et al., 1996a), as well as for the automatic construction of fuzzy controllers for the complex biochemical process of penicillin production (Araúzo et al., 1999; Cano, Dimitriadis, Araúzo, Abajo & Coronado, 1996b; Gómez, Cano, Araúzo, Dimitriadis & Coronado, 1998b) and for the traffic control over an ATM network (Custodio, Tascón, Merino & Dimitriadis, 1999).

2.5. FasArt algorithm

This section presents FasArt algorithm in a comprehensive step-by-step format. We separate the training and testing stages, although incremental learning performance can be achieved, by allowing FasArt weights to be updated according to the learning laws during the testing stage. The newly introduced activation/membership function is used for calculating the activity of each neuron both in the training and testing stages. However, in the training stage a winner-takes-all (WTA) mechanism is applied so that only one category is allowed to learn. On the contrary, in the prediction stage all active units in F_2^a can contribute to the output through the inter-ART weights, as reflected by the defuzzification function, since FasArt can be seen as a fuzzy logic system during this stage.

2.5.1. FasArt training

During FasArt training, input pairs ($\mathbf{a}^1$, $\mathbf{b}^1$), ($\mathbf{a}^2$, $\mathbf{b}^2$), …, ($\mathbf{a}^n$, $\mathbf{b}^n$)… are presented. Prior to training, there are not committed units or relations stored in the inter-ART map. Then for each training pair:

Step 1—Calculation of complementary code: vector $\mathbf{I^a}$ is formed by

$$\mathbf{I^a} = (\mathbf{a}, \mathbf{a}^c) = (\mathbf{a}_1, \ldots, \mathbf{a}_M, \mathbf{a}_1^c, \ldots, \mathbf{a}_M^c)$$

Similarly vector $\mathbf{I^b}$ is formed from $\mathbf{b}$.

Step 2—Calculation of activation in F_2^a and F_2^b: for each unit j in F_2^a, its activation η_{R_j} is given by

$$\eta_{R_j} = \prod_{i=1}^{M} \eta_{ji}(I_i^a)$$

where function η_{ji} is defined by:

$$\eta_{ji}(I_i) = \begin{cases} \max\left(0, \dfrac{\gamma_a(I_i^a - w_{ji}^a) + 1}{\gamma_a(c_{ji}^a - w_{ji}^a) + 1}\right) & \text{if } I_i^a \le c_{ji}^a \\ \max\left(0, \dfrac{\gamma_a(1 - I_i^a - w_{ji}^{ca}) + 1}{\gamma_a(1 - c_{ji}^a - w_{ji}^{ca}) + 1}\right) & \text{if } I_i^a > c_{ji}^a \end{cases}$$

Activity in F_2^b units is computed in a similar way.

Step 3a—Select winner units in F_2^a: a WTA mechanism is applied at the F_2^a layer to select the winner node J, which is that of maximal activation, i.e. $\eta_{R_J} = \max_j \eta_{R_j}$. If all units are inhibited, a new unit is committed.

Step 3b—Select winter units in F_2^b: the same process is independently carried out in F_2^b to find winner K. The order of steps 3a and 3b is unimportant (ideally, they could be carried out in parallel in a hardware implementation).

Step 4a—Reset evaluation in F_2^a: for the winner J, if

$$\frac{|\mathbf{I^a} \wedge \mathbf{W_J^a}|}{|\mathbf{I^a}|} < \rho_a$$

the reset occurs. Unit J is inhibited for the rest of this presentation ($\eta_{R_J} = -1$). Go again to **Step 3a**.

Step 4b—Reset evaluation in F_2^b: the same process is carried out in F_2^b. If reset occurs, go to **Step 3b**.

Step 5—Evaluate inter-ART matching: the inter-ART reset occurs if

$$w_{JK}^{ab} \neq 1$$

which means that unit J in F_2^a does not predict K in F_2^b. If inter-ART reset is fired, unit J in F_2^a is inhibited, and ρ_a is raised temporarily by

$$\rho_a = \frac{|\mathbf{I^a} \wedge \mathbf{W_J^a}|}{|\mathbf{I^a}|}$$

and a new unit is selected (go to **Step 3a**, but **Step 3b** is not performed again).

Step 6—Learning: learning is carried out. In each unsupervised module, learning follows the general laws:

$$\mathbf{W_J^{a(new)}} = \beta_a(\mathbf{I^a} \wedge \mathbf{W_J^{a(old)}}) + (1 - \beta_a)\mathbf{W_J^{a(old)}}$$

$$\mathbf{C_J^{a(new)}} = \beta_a^c \mathbf{I^a} + (1 - \beta_a^c)\mathbf{C_J^{a(old)}}$$

$$\mathbf{W_K^{b(new)}} = \beta(\mathbf{I^b} \wedge \mathbf{W_K^{b(old)}}) + (1 - \beta_b)\mathbf{W_K^{b(old)}}$$

$$\mathbf{C_K^{b(new)}} = \beta_b^c \mathbf{I^b} + (1 - \beta_b^c)\mathbf{C_K^{b(old)}}$$

If unit J in ARTa is newly committed, fast learning is performed (i.e. $\beta_a = \beta_a^c = 1$). The same holds for unit K in ARTb.

The inter-ART map is modified, so that $w_{JK}^{ab} = 1$, and all other $w_{Jk} = 0$, for $k = 1, \ldots, M$, $k \neq K$.

2.5.2. FasArt testing

During the prediction stage, FasArt can be seen as a fuzzy logic system. It receives test patterns $\mathbf{a}^1$, $\mathbf{a}^2$, …, $\mathbf{a}^n$,… to which FasArt must associate predicted outputs. For each input pattern:

Step 1—Calculation of complementary code: input vector $\mathbf{I^a}$ is calculated from $\mathbf{a}$, as shown in training, step 1.

Step 2—Calculation of activation in F_2^a: the activation η_{R_k} is calculated as described during training, step 2. This can be interpreted as finding the fuzzy membership degree of the input pattern to the different fuzzy sets stored as antecedents in the rule base.

Step 3—Inference: find for each of the active units j in F_2^a its corresponding unit in F_2^b through the inter-ART map. This can be seen as applying the compositional rule of inference (see Definition 11, in Section 2.1).

This step outputs several fuzzy sets in the output space, defined by the weights of F_2^b units, and the membership degree produced by the input pattern in the antecedent of their rules. Two approaches can be followed to obtain the final output (Lee, 1990).

Step 4—Defuzzification: for FasArt the defuzzification has been calculated as the average of fuzzy sets centers, as given by

$$y_m(\mathbf{I}) = \frac{\sum_{k=1}^{N^b} \sum_{j=1}^{N^b} c^b_{km} w^{ab}_{jk} \eta^a_{R_j}(\mathbf{I})}{\sum_{k=1}^{N^b} \sum_{j=1}^{N^b} w^{ab}_{jk} \eta^a_{R_j}(\mathbf{I})}$$

Then the output is a real value in the output space. This is the approach followed in this paper.

Step 4′—Fuzzy labeling: if the output fuzzy sets can be labeled with linguistic variables (e.g. cold, warm, hot), we may be interested in retaining to which degree these fuzzy sets were inferred, instead of finding a crisp output. This approach could be followed to apply FasArt for pattern recognition.

2.6. Enhancing performance with prediction error minimization with FasBack

During our experimental studies with FasArt, it was observed that the network complexity, expressed in terms of number of units, was rather high for a given prediction error. Therefore, we proposed the FasBack neuro-fuzzy system in order to reduce system complexity, while having a guarantee of the same performance, or equivalently to enhance performance with the same system complexity. The new significant element of FasBack refers to learning guided by prediction error minimization, besides the basic mechanism of learning by pattern matching that is employed in FasArt and in all previous models of the ART family. Thus, the new objective of FasBack is the mathematic minimization of the prediction error:

$$\mathbf{e} = \mathbf{y} - \mathbf{d} \tag{13}$$

where $\mathbf{y}$ denotes the system output and $\mathbf{d}$ the desired output.

The corresponding error index that has to be minimized is given by the following expression:

$$I = \frac{1}{2}\sum e_i^2 = \frac{1}{2}\sum (y_i - d_i)^2 \tag{14}$$

Then, our goal consists in choosing the appropriate system parameters that minimize the above error index. We can easily observe that we face an optimization problem of quadratic indices for nonlinear systems, that can be tackled with any of the methods proposed in the related technical literature. The gradient descent method has been selected, in which parameters are modified in the direction indicated by the derivative of the error index with respect to the parameter vector (Wang, 1994), that is:

$$\mathbf{p}(k) = \mathbf{p}(\mathbf{k}-1) - \epsilon \left.\frac{\partial \mathbf{e_p}}{\partial \mathbf{p}}\right|_{\mathbf{k}-1} \tag{15}$$

where $p(k)$ denotes the parameter vector with respect to which we optimize at instant k, $0 \leq \epsilon \leq 1$ is a scalar that represents learning rate and e_p is the prediction error given by the following expression:

$$\mathbf{e_p} = \frac{1}{2}(\mathbf{y} - \mathbf{d})^2 \tag{16}$$

If we derive the above expression with respect to system parameter vector $\mathbf{p}$ we obtain

$$\frac{\partial \mathbf{e_p}}{\partial \mathbf{p}} = (\mathbf{y} - \mathbf{d})\frac{\partial \mathbf{y}}{\partial \mathbf{p}} \tag{17}$$

Introducing (17) into (15), we obtain the following learning law for the system parameters:

$$\mathbf{p(k)} = \mathbf{p}(\mathbf{k}-1) - \epsilon(\mathbf{y} - \mathbf{d})\left.\frac{\partial \mathbf{y}}{\partial \mathbf{p}}\right|_{\mathbf{k}-1} \tag{18}$$

There exist several alternatives for the parameter vector to be optimized. The first refers to $\mathbf{p} = (\mathbf{c^b_{jm}})$, i.e. in this case the optimization is made to the shape of the fuzzy sets in ARTb, through their centers. The corresponding expression is:

$$c^b_{jm}(t+1) = c^b_{jm}(t) - \epsilon(y_m - d_m)\frac{\sum_{k=1}^{N^a} w^{ab}_{kj}\eta_{R_k}(\mathbf{I})}{\sum_{l}^{N^b}\sum_{k=1}^{N^a} w^{ab}_{kl}\eta_{R_k}(\mathbf{I})} \tag{19}$$

As a second option, the optimization can be applied to $\mathbf{p} = (\mathbf{w^{*ab}_{ij}})$, i.e. the connections that form a rule. These weights of inter-ART map $w^{*ab}_{ij} = w^{ab}_{iJ}$ correspond to a unique unit J where $w^{ab}_{ij} \neq 0$, as it was defined in FasArt. In this case, the learning law is:

$$w^{*ab}_{ij}(t+1) = w^{*ab}_{ij}(t) + \epsilon(y_m - d_m)(y_m - c^b_{jm})\frac{\eta_{R_i}(\mathbf{I})}{\sum_{l=1}^{N^b}\sum_{k=1}^{N^a} w^{ab}_{kl}\eta_{R_k}(\mathbf{I})} \tag{20}$$

Finally, the form of the input fuzzy sets in ARTa can be optimized through their centers $\mathbf{p} = (\mathbf{c}^{\mathbf{i}}_{\mathbf{an}})$, thus obtaining the following learning law:

$$c^a_{in}(t+1) = c^a_{in} + \epsilon(y_m - d_m)\frac{\eta_{R_i}(\mathbf{I})}{\eta_{in}(I_n)}\frac{\partial\eta_{in}(I_n)}{\partial c^a_{in}}\frac{\sum_{l=1}^{N^b} w^{ab}_{il}(y_m - c_{lm})}{\sum_{l=1}^{N^b}\sum_{k=1}^{N^a} w^{ab}_{kl}\eta_{R_k}(\mathbf{I})} \quad (21)$$

where:

$$\frac{\partial\eta_{in}(x_n)}{\partial c_{in}} = \begin{cases} \frac{-(\gamma(x_n - w_{in}) + 1)\gamma}{(\gamma(c_{in} - w_{in}) + 1)^2} & \text{if } w_{in} - \frac{1}{\gamma} < x_n \le c_{in} \\ \frac{(\gamma(1 - x_n - w^c_{in}) + 1)\gamma}{(\gamma(1 - c_{in} - w^c_{in}) + 1)2} & \text{if } c_{in} < x_n < 1 - w^c_{in} + \frac{1}{\gamma} \\ 0 & \text{otherwise} \end{cases} \quad (22)$$

It can be easily shown that if new elements are added to the parameter vector, then the number of degrees of freedom in the system is increased, and, therefore, further reduction of prediction error is possible with the cost of additional computational requirements. Thus, the best results can be obtained with a parameter vector that combines the aforementioned three alternatives, i.e. with $\mathbf{p} = (\mathbf{c}^{\mathbf{b}}_{\mathbf{jm}}, \mathbf{w}^{*\mathbf{ab}}_{\mathbf{i}}, \mathbf{c}^{\mathbf{a}}_{\mathbf{jm}})$ as shown in Cano et al. (1997). The following remarks can be made about the global optimization process and the specific formulas.

1. Although learning equations were formulated as a result of applying a mathematical optimization method, we have seen that it is possible to have an interpretation in terms of 'natural language'.
2. All previous formulas have no mathematical sense if activation of all systems units is null, i.e. if $\eta_{R_k}(\mathbf{I}) = \mathbf{0}$ for every unit k. Therefore, prior to this error minimization learning law, another learning law should be applied, that can guarantee a sufficient spanning of the input space by the unit weights. This step can be performed by the usual pattern matching learning law of FasArt. Then, for each input pattern a learning cycle using the FasArt pattern matching algorithm is performed first, followed by a second cycle in which system parameters are modified according to the previous prediction error minimization formulas.
3. Parameter variation is null for all units or relations that are not active for a certain input vector, i.e. for which $\eta_{R_k}(\mathbf{I}) = \mathbf{0}$. We then have a learning process that affects only the relations (rules) activated by the input vector. In other terms, we perform a piecewise error minimization process instead of global minimization error, as in traditional multilevel perceptrons with backpropagation learning. In our system, we search for subvectors of the parameter vector, that locally minimize prediction error for a certain variation of the input. This property implies that incremental, on-line learning is still valid, as in previous FasArt and Fuzzy ARTMAP architectures.

3. Learning with noisy patterns: performance and comparative evaluation

New significant problems appear when we deal with real-world experimental data. Among them, we can emphasize the presence of a certain degree of noise that is caused by diverse sources, such as system perturbations or uncertainties, in the measurement instrumentation.

The presence of noise in training data adds an important problem, since it affects the generalization capability of the neural architecture and forces modifications of the error index to be optimized (Bishop, 1994). On the other hand, fuzzy systems can be considered as appropriate candidates for this type of problem, since they establish an imprecise separation among classes. In this section, we present the most significant solutions that were proposed with respect to learning from noisy patterns within ART family neuro-fuzzy architectures.

Williamson (1996) claimed the following deficiencies of Fuzzy ARTMAP in handling the above problem:

Sensibility to noise: the inter-ART reset mechanism produces category proliferation. Carpenter, Grossberg and Reynolds (1995) propose the use of slow learning as well as an a priori maximum number of categories, and more recently the use of distributed learning in dARTMAP (Carpenter et al., 1998) as a means to remedy this problem.

Use of inadequate fuzzy categories: Fuzzy ARTMAP uses fuzzy categories of hyperrectangular form, while representations based on hyperspheres seem to be more adequate to handle noisy patterns.

Three general strategies were proposed in order to solve these initial deficiencies of Fuzzy ARTMAP.

1. The learning law related to the modification of weights in inter-ART map should be modified. Then, these weights should tend to represent the relationship between categories in the two Fuzzy ART modules. In this sense, ART-EMAP (Carpenter & Ross, 1995; Rubin, 1995) incorporates slow learning of inter-ART map weights and establishes that, when an a priori maximum number of categories is reached in F^a_2, inter-ART reset mechanism should not fire again. Furthermore, an evidence accumulation method can be used when a more difficult classification decision cannot be made. A similar methodology is employed in PROBART (Marriott & Harrison, 1995; Srinivasa, 1997), where the inter-ART map weights count the number of times that two categories were associated.

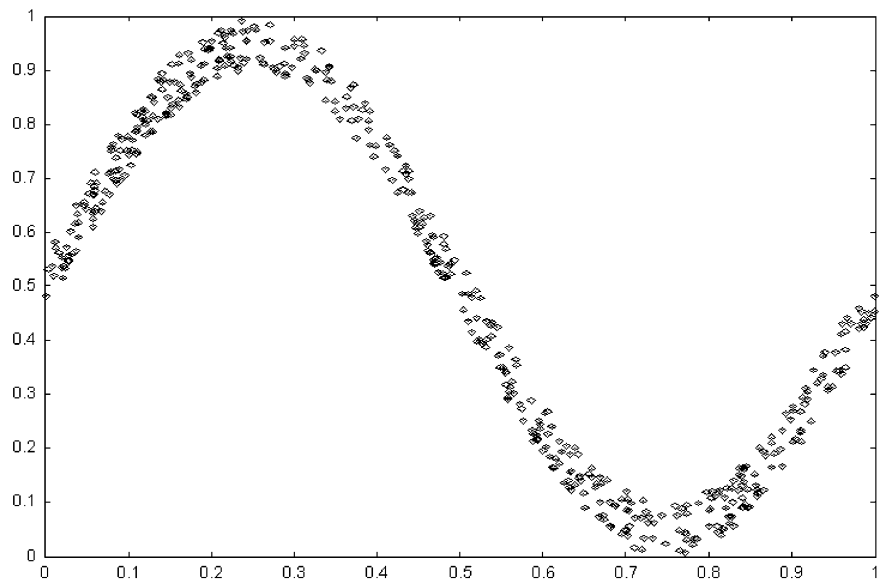

Fig. 4. Noisy patterns of the initial function to be approximated, given by $f(x) = 0.45(1 + \sin(2\pi x))$, with additive noise from a uniformly distributed source in the interval $[-0.1, 0.1]$.

2. Fuzzy categories should be represented in an appropriate way, according to data statistics, typically of Gaussian nature. In this sense, Gaussian ARTMAP was proposed (Williamson, 1996), following the general principles of Gaussian (Firmin & Hamad, 1994) or radial basis function neural networks (Chen & Chen, 1995; Jokinen, 1992; Mulgrew, 1996; Osman & Fahmy, 1994; Roy, Govil & Miranda, 1995).
3. Use of a distributed codification (Carpenter et al., 1998) which, as opposed to winner-takes-all mechanism, allows several units to contribute to the output.

All previous models were applied in pattern classification problems, except PROBART that was also used for function approximation. Since the main goal of this paper is to study the relative merits of FasArt and FasBack in function approximation problems with noisy learning patterns, it is reasonable to include PROBART as a reference point for the experimental study. In this sense, two performance indices are considered, the prediction error (RMSE) and the number of committed categories. Results for Fuzzy ARTMAP and PROBART are taken from Marriott and Harrison (1995), and FasArt parameters are tuned based on experience to produce a similar result in one of the performance indices in order to establish a fair comparison on the other index.

In addition, it is interesting to have a reference point in some well-known statistical technique. In particular, we have compared the neural systems to kernel smoothing methods (Bickel & Rosenblatt, 1973).

3.1. Function approximation with FasArt and FasBack

In order to evaluate the capabilities of FasArt and FasBack to approximate functions, when learning patterns are noisy, let us use the function $f(x) = 0.45(1 + \sin(2\pi x))$, when a uniformly distributed additive noise ϵ in the interval $[-0.1, 0.1]$ is present in the learning patterns. Fig. 4 shows the 500 learning patterns that were randomly generated in $x \in [0, 1]$.

Using the above learning patterns, both FasArt and FasBack models were trained and tested with the following design parameters: $\rho_a = 0.66$, $\rho_b = 0.5$, $\gamma_a = \gamma_b = 10$, $\beta_a = \beta_b = 1$, and for FasBack $\epsilon = 0.4$. As previously mentioned, these parameters have been selected for a fair comparison to the other architectures. As can be seen from Fig. 5, both models perform well in this problem, although FasBack shows a major capability for generalization for the same network complexity.

To assess the influence of noise in the results of both models, they were also trained with non-noisy patterns. Furthermore, a widely accepted quantitative measure, such as RMSE (Relative Mean Square Error), has been used in order to draw valid conclusions. In the results, shown in Table 2, it can be seen that FasBack is better than FasArt in both cases for equivalent network complexity. It is noteworthy that difference in performance (prediction error) is not significant when noise is incorporated into the learning patterns. It can therefore be concluded that

Table 2
Summary of function approximation results using FasArt and FasBack with noisy or non-noisy learning patterns

Model	Presence of noise	Number of rules	RMSE
FasArt	Yes	11	0.0455
FasBack	Yes	11	0.0158
FasArt	No	13	0.0336
FasBack	No	15	0.0056

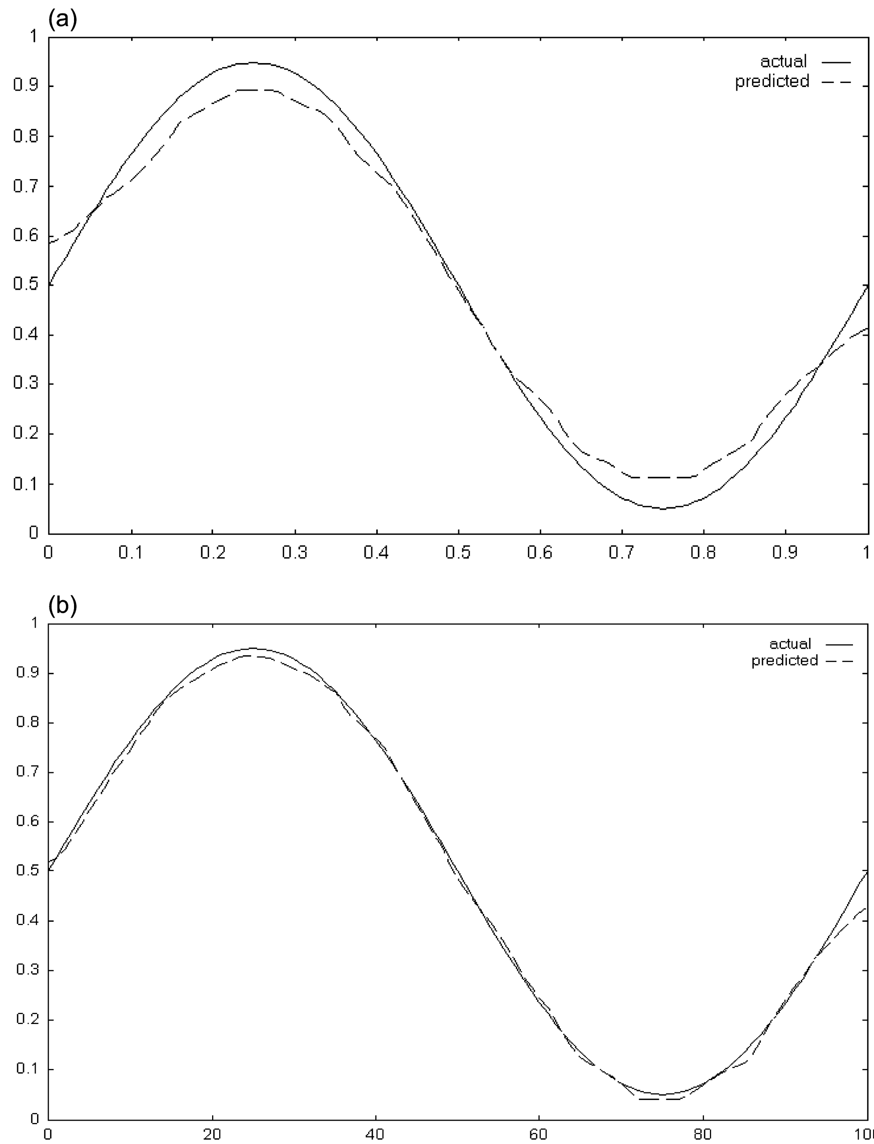

Fig. 5. Performance of (a) FasArt and (b) FasBack in approximating a sinusoidal function, using noisy learning patterns, where real is plotted solid, and predicted is dashed.

both models perform well when noise is present during the learning phase, although this feature was not explicitly taken into account during the design of these models. Finally, it cannot be argued that FasBack is less sensible to perturbations caused by the noisy learning patterns, since it is significantly better even in the case of non-noisy learning patterns.

3.2. Experimental comparative performance study of Fuzzy ARTMAP, PROBART, FasArt and FasBack

PROBART is the only ART model that has been explicitly designed for and applied to the problem of learning using noisy patterns. It follows the same basic architecture and algorithm as Fuzzy ARTMAP, with a modified activation function for the inter-ART map, where:

$$\mathbf{X}^{\mathbf{ab}} = \begin{cases} \mathbf{Y}^{\mathbf{b}} + \mathbf{W}_{\mathbf{J}}^{\mathbf{ab}} & \text{if unit } J \text{ of } F_2^a \text{ is active and } F_2^b \text{ is active} \\ \mathbf{W}_{\mathbf{J}}^{\mathbf{ab}} & \text{if node } J \text{ of } F_2^a \text{ is active and } F_2^b \text{ is not active} \\ \mathbf{Y}^{\mathbf{b}} & \text{if } F_2^a \text{ is not active and } F_2^b \text{ is active} \\ 0 & \text{if neither } F_2^a \text{ nor } F_2^b \text{ are active} \end{cases} \quad (23)$$

Comparing it with Eq. (6), it can be seen that the only difference consists in the replacement of fuzzy intersection operator $\wedge$ by the sum operator $+$. Learning in inter-ART map is also performed with a fast learning law, i.e.

$$\mathbf{W}_{\mathbf{J}}^{\mathbf{ab(new)}} = \mathbf{X}^{\mathbf{ab}} \quad (24)$$

Therefore, weights in inter-ART map reflect the number of times that a certain association has been activated due to a new learning input pattern. According to the Hebbian principles of learning, connections with more activation correspond to those of a greater value of the weight.

In PROBART, the mechanism of inter-ART reset is not used and, therefore, the vigilance parameter remains constant, or equivalently, the size of categories in Fuzzy ARTa is kept constant. Due to this fact, frequency of associations between categories remains unchanged. In a similar way, designers of PROBART do not use inter-ART vigilance parameter ρab, thus allowing one-to-many relations among nodes in ARTa and ARTb, where weights (frequencies) reflect the importance of each relation. Finally, a new equation is proposed for calculating the output, that takes into account the values of frequencies accumulated in the inter-ART map weights:

$$\mu_{Jm} = \frac{1}{|\mathbf{W_J^{ab}}|}\sum_{n=1}^{N_b} \epsilon_{nm} w_{Jn}^{ab}, \qquad m = 1, \ldots, 2M_b \tag{25}$$

where μ_{Jm} is the predicted value for the m-th component of the output vector associated with node J of ARTa, $|\mathbf{W_J^{ab}}|$ is the total number of associations between nodes in ARTb and node J in ARTa, ϵ_{nm} is the m-th component of the vector associated with the n-th category in ARTb, and w_{Jn}^{ab} is the frequency associated with the n-th category of ARTb and node J of ARTa.

It can be concluded that the main difference between PROBART and Fuzzy ARTMAP is the suppression of inter-ART reset and the corresponding mechanism of match tracking. Then, in a PROBART architecture, classification in ARTa is performed in a totally unsupervised way, as in ARTb, as opposed to Fuzzy ARTMAP, where generation of categories in ARTa is guided by the equivalent process in ARTb.

Exception handling in PROBART is then included in the general mechanism of reflecting the frequency of associations between categories in the inter-ART map weights, since a low frequency indicates an exception. On the contrary, in Fuzzy ARTMAP a contradiction between past knowledge and a new input pattern generated a new category for an exception. Although the mechanism adopted in PROBART for exception handling avoids category proliferation, it may produce undesired performance, especially in classification problems. There exist many cases, where there are not many learning patterns corresponding to an exception situation, that in turn may have a great importance in test phase. For example, in the problem of poisonous mushrooms described in Carpenter et al. (1991e), there may be just a few exemplars of poisonous mushrooms that may share several common features with a category of edible mushrooms with many exemplars in the learning set. If a totally unsupervised classification is performed in ARTa, as proposed in PROBART, it is possible that both types of mushroom fall into the same class. Then, the predicted value in the test phase will always correspond to an edible mushroom, because of the greater percentage of such mushrooms within this class. Inter-ART reset mechanism solves the above problem, creating two categories, since class formation in ARTa is guided by the supervision information provided in ARTb.

In order to evaluate the relative performance of FasArt, FasBack neuro-fuzzy systems and PROBART architecture, a function approximation test proposed by Marriott and Harrison (1995) is carried out. The function to be approximated is:

$$f(x) = \frac{\sum_{k=1}^{7} \sin(10kx) + 10}{20} \tag{26}$$

The range of test for $f : \Re \rightarrow \Re$ is $f(x) \in [0.2295, 0.7705]$ for an input domain of $x \in [0, 1]$. Noisy patterns are generated by adding Gaussian noise to the original patterns according to:

$$y_p = f(x_p) + 0.02\epsilon_p \tag{27}$$

where ϵ_p is Gaussian noise $\epsilon_p \sim \mathcal{N}(0, 1)$.

Following the methodology proposed in Marriott and Harrison (1995), 1000 non-noisy learning patterns were generated with randomly selected points x, that can be observed in Fig. 6. Input and output vectors were normalized to [0, 1] for all the architectures studied in this section. FasArt and FasBack models were trained using the same design parameters: $\rho_a = 0.5$, $\rho_b = 0.9$, $\gamma_a = \gamma_b = 50$, $\beta_{a-} = \beta_b = 1$, and for FasBack $\epsilon = 0.4$. For the test phase, another set of 1000 randomly generated patterns was chosen. Fig. 7 shows the results, as well as the prediction error for FasArt and FasBack. It can be easily seen that identification is very good in both cases and that error is kept low and without apparent structure.

In Marriott and Harrison (1995) Fuzzy ARTMAP and PROBART were tested for the same experiment and under the same conditions, using the following design parameters: $\alpha = 0.001$, $\rho_a = 0.99$, $\rho_b = 0.99$, $\rho_{ab} = 0.9$. All these results are collected in Table 3, where *Na* and *Nb* represent the number of nodes, that were required at ARTa and ARTb, and MAXE denotes the maximum value of the absolute prediction error.

It can be easily appreciated in Table 3, that FasBack and FasArt are clearly better than Fuzzy ARTMAP in the task of approximating this function, since the index of quadratic error RMSE is smaller, also with a considerably smaller

Table 3
Comparison of results for the models under study when non-noisy learning patterns were used in order to identify the benchmark function

Model	*Na*	*Nb*	RMSE	MAXE
ARTMAP	312	53	0.0074	0.01
PROBART	110	53	0.0169	0.0755
FasArt	130	29	0.0066	0.0425
FasBack	133	29	0.0035	0.0327

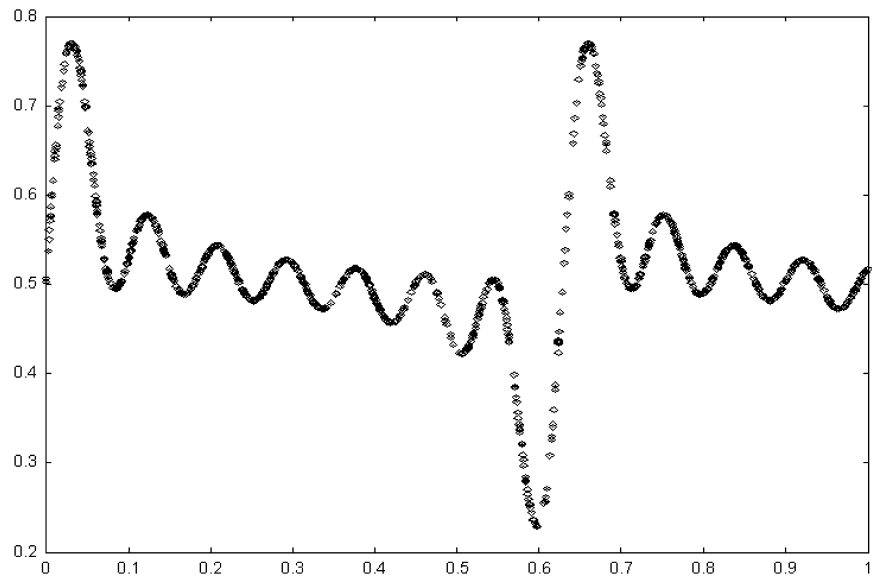

Fig. 6. Non noisy patterns of the benchmark function to be approximated, given by Eq. (26) (Marriott & Harrison, 1995).

network complexity, measured as number of nodes. With respect to PROBART, both FasArt and FasBack present better error indices with an equivalent network complexity.

The second experiment, proposed in Marriott and Harrison (1995), deals with the same task but using noisy learning patterns. A new learning set of 1000 patterns was generated, using Eqs. (26) and (27). Once learning was performed with the data shown in Fig. 8, the test was carried out using 1000 non-noisy patterns that were chosen randomly.

Fig. 9 shows the predicted function, as well as the prediction error, when FasArt and FasBack were used with the same design parameters as in the previous experiment. An easy qualitative conclusion can be drawn from this figure, i.e. that FasArt and FasBack are capable of representing such a bench-mark function, even when learning data are corrupted. Such a property is especially important in real-world applications of system identification, where data coming from sensors present a considerable noise level. An identification model that is insensitive to the presence of noise in the learning phase is very useful since special data preprocessing techniques can then be avoided.

Results for Fuzzy ARTMAP and PROBART in this experiment are collected in Table 4 according to conditions imposed in Marriott and Harrison (1995) and keeping the same parameters: α, ρ_a, ρ_b and ρ_{ab}. Error data that are denoted by legend (L) refer to results obtained for the learning set. On the other hand, those denoted with legend (CL) refer to results obtained when the test set coincided with the learning set, but with the significant difference that no Gaussian noise was present, i.e. using Eq. (26) instead of Eq. (27) for the output points. Thus, with this experiment, we may find out the capabilities of the different models to 'filter out' noise present in learning patterns. Finally, error data denoted by legend (T) refer to the normal test set.

Analyzing data in Table 4, it can be easily observed that the presence of noise in the learning set of Fuzzy ARTMAP produces a considerable increase of **number of nodes** *Na* in ARTa (from 312 to 806). This is due to the fact that inter-ART reset mechanism fires incorrectly, since two points, that should belong to the same category, are now part of different categories because of the presence of noise.

To illustrate such behavior, consider two points x_1 and x_2 that belong to the same category in ARTa. When we perform learning using non noisy patterns, their corresponding points belong to the same category in ARTb, i.e. $y_1 = f(x_1)$ and $y_2 = f(x_2)$ according to Carpenter et al., (1991c)should be separated by a distance R so that:

$$R \leq M(1 - \rho) \tag{28}$$

For this case $M = 1$ and $R = |1 - w_2^i - w_1^i|$ where $w = (w_1^i, w_2^i)$ is the weight vector associated with class i to which $f(x_1)$ and $f(x_2)$ belong. Then, for fast learning of non noisy patterns:

$$|y_1 - y_2| = |f(x_1) - f(x_2)| \leq M(1 - \rho) \tag{29}$$

On the other hand, when using noisy patterns in the learning set, the corresponding output points are now $f(x_1) + \epsilon_1$ and $f(x_2) + \epsilon_2$, and therefore it is possible that the following inequality be true:

$$|y_1 - y_2| = |f(x_1) + \epsilon_1 - f(x_2) - \epsilon_2| > M(1 - \rho) \tag{30}$$

In this case, y_1 and y_2 do not belong to the same category in ARTb, and therefore corresponding points x_1 and x_2 cannot be assigned to the same category in ARTa.

This fact can explain the considerable increase of categories in ARTa, as compared with the small increase of nodes N_b in ARTb (from 53 to 61), produced by the augmented range of $y(x)$ due to addition of noise. On the other hand, PROBART does not present such a category proliferation problem (from

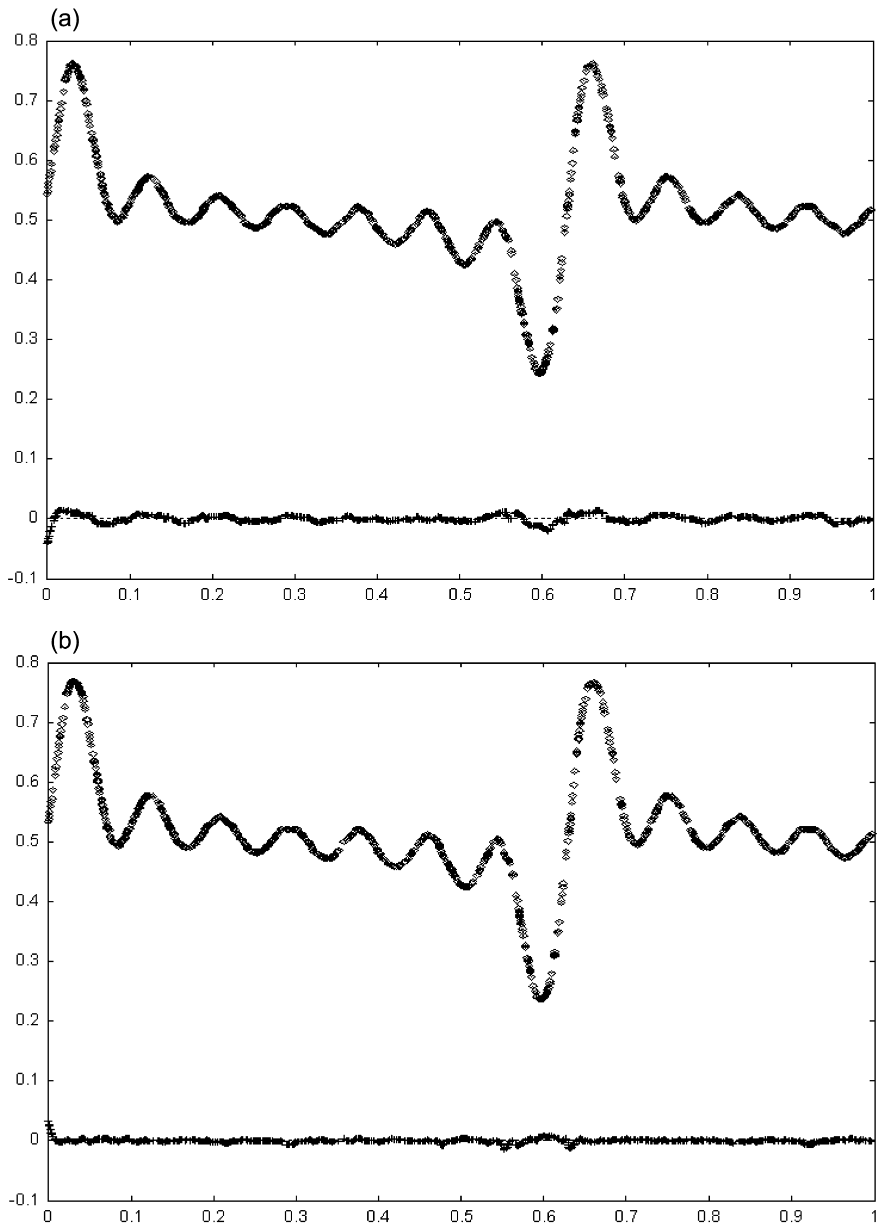

Fig. 7. Performance of (a) FasArt and (b) FasBack in approximating a benchmark function, using non noisy learning patterns. Top plot corresponds to the identification performed by the model, while bottom plot is the prediction error.

100 to 112 categories) due to the modification in the inter-ART reset mechanism. Performance in ARTb is similar to that of Fuzzy ARTMAP, since both perform self-organizing learning. As far as FasArt and FasBack results are concerned, they are closer to those of Fuzzy ARTMAP, since the same inter-ART reset mechanism and match tracking procedure are kept in their design. However, the increase of nodes in ARTa and ARTb is much smaller than that in Fuzzy ARTMAP, since FasArt and FasBack can work with lower vigilance parameters in order to obtain sufficient category discrimination ($\rho_a = 0.5$, $\rho_b = 0.9$, as compared with $\rho_a = \rho_b = 0.99$ in Fuzzy ARTMAP).

The **performance characteristics** due to the presence of noise in the learning patterns can also be analyzed from Tables 3 and 4. In Fuzzy ARTMAP, error index RMS has significantly deteriorated (from 0.0074 to 0.0302), although the number of categories increased. On the other hand, error RMS(L) is much less for the learning set (0.0137), as compared with both cases of test RMS(CL) and RMS(T) (0.0302). It can be deduced that Fuzzy ARTMAP has performed 'overlearning', i.e. it overfitted learning data and, therefore, it was not able to filter out noise and generalize for the new data of the test set. It can be argued that Fuzzy ARTMAP has tried to learn noise besides signal information. Therefore, it generated many categories that contained contradictory information, since they were mainly due to noise and not to significant information of input patterns.

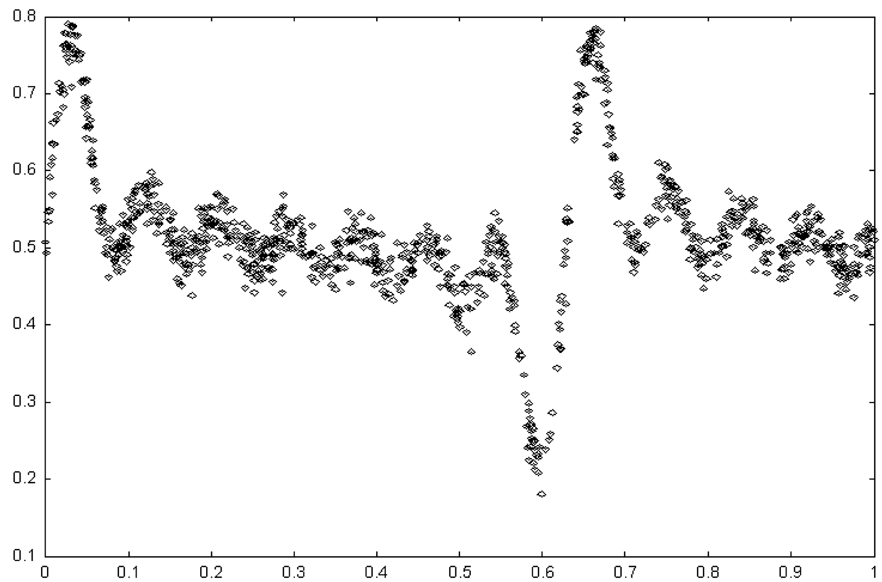

Fig. 8. Noisy patterns of the benchmark function to be approximated, given by Eq. (26) (Marriott & Harrison, 1995), where the output is corrupted by additive Gaussian noise, as shown by Eq. (27).

With respect to PROBART, error is higher for the case of learning with noisy patterns (from 0.0168 to 0.0202), as expected, but this increase is much smaller than that of Fuzzy ARTMAP. Another significant observation is that error is higher in learning set error RMS(L) = 0.0322 than in both test sets (RMS(CL) = 0.0189, RMS(T) = 0.0202). It can then be deduced that PROBART is capable of filtering out noise present in the learning set, and it can better learn the basic signal that generated learning data. Finally, it is apparent that both FasArt and FasBack produce much better error indices in both test sets, enhancing results provided by PROBART, with a cost of an increase in the number of categories.

In order to test the validity of neural approaches, we can compare them to other well-known statistical techniques. In particular, kernel regression (Bickel & Rosenblatt, 1995) has been used for the estimation of function given by Eq. (26). We have used a bandwidth of 1% of the range of the input variable, and quartic kernel functions. If non noisy observations are used for the regression kernels (Table 3), RMS = 0.0049 and MAXE = 0.0483 are obtained. If noisy observations are used, RMS(L) = 0.0197 and MAXE(L) = 0.0689; RMS(CL) = 0.0069 and MAXE(CL) = 0.0400; RMS(T) = 0.0073 and MAXE(T) = 0.0375. These results are comparable to those achieved by FasArt and FasBack, while these neuro-fuzzy systems achieve code compression, and feature several important properties such as fast incremental learning, that make them suitable for on-line adaptive applications. In addition, FasArt and FasBack also permit interpretation of the acquired knowledge as fuzzy rules.

In Marriott and Harrison (1995) another experiment was proposed in order to enhance error indices. In order to achieve this objective, an increase in the number of learning patterns (10,000 instead of 1000) was proposed, and the values of the vigilance parameters were raised ($\rho_a = \rho_b = 0.998$). Observing the results shown in Table 5, it can be seen that the number of categories increased, as compared with the previous experiment. Comparing these results with those obtained by FasBack in the previous experiment, it is clear that our model achieves the same error indices with a much lower number of categories and with much less learning patterns.

An interesting design property of FasBack is that a smaller number of rules is obtained by error-based learning, because it can now work with lower vigilance parameters. To test this, the initial learning set of 1000 noisy patterns was used with the design parameters ($\rho_a = 0.5$, $\rho_b = 0.9$, $\gamma_a = \gamma_b = 50$. Observing the results shown in Table 6, it can be seen that a significant reduction in the number of nodes was achieved, while keeping the same magnitude of

Table 4
Comparison of results for the models under study when noisy learning patterns were used in order to identify the benchmark function

Model	*Na*	*Nb*	RMS(L)	RMS(CL)	RMS(T)	MAXE(L)	MAXE(CL)	MAXE(T)
F. ARTMAP	806	61	0.0137	0.0302	0.0302	0.0878	0.0678	0.0679
PROBART	112	61	0.0322	0.0189	0.0202	0.1507	0.0769	0.0905
FasArt	275	30	0.0206	0.0099	0.0097	0.0716	0.0411	0.0427
FasBack	284	30	0.0198	0.0074	0.0078	0.0810	0.0364	0.0389

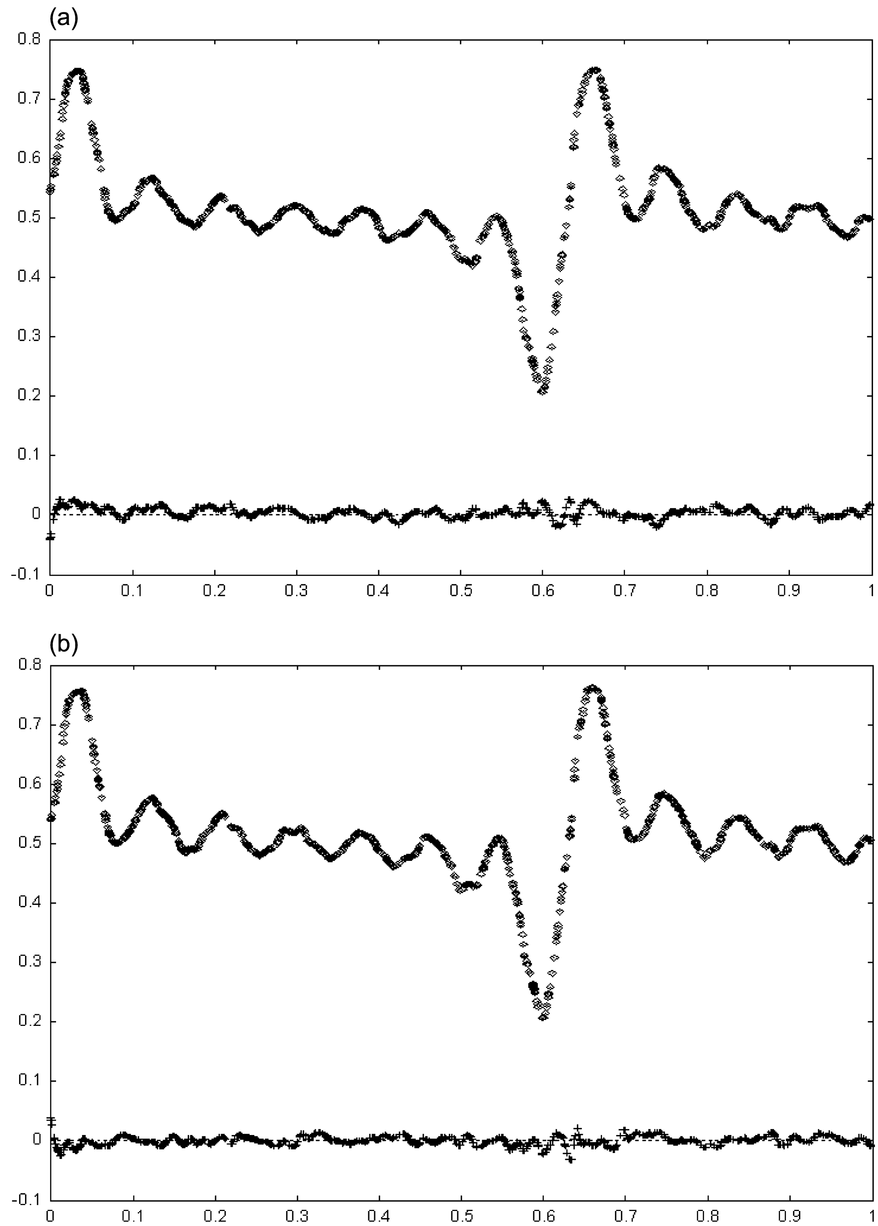

Fig. 9. Performance of (a) FasArt and (b) FasBack in approximating a benchmark function, using noisy learning patterns. Top plot corresponds to the identification performed by the model, while bottom plot is the prediction error.

Table 5
Best effort results of PROBART for noisy learning patterns when using increased vigilance parameters and size of learning set

Model	*Na*	*Nb*	RMS(L)	RMS(CL)	RMS(T)	MAXE(L)	MAXE(CL)	MAXE(T)
PROBART	608	341	0.0276	0.0079	0.0084	0.0779	0.0219	0.0269

Table 6
Results obtained by FasBack with lower vigilance parameters and the initial learning set

Model	*Na*	*Nb*	RMS(L)	RMS(CL)	RMS(T)	MAXE(L)	MAXE(CL)	MAXE(T)
FasBack	88	15	0.0209	0.0096	0.0098	0.0795	0.0458	0.0481

Table 7
Structural and algorithmic comparison between Fuzzy ARTMAP, PROBART, FasArt and FasBack

Item	Fuzzy ARTMAP	PROBART	FasArt	FasBack
Duality to fuzzy logic system	No		Yes	
Inter-ART	Competitive	Probabilistic	Competitive	
Output prediction	Competitive		Distributed	
Inter-ART reset	Yes	No	Yes	
Unit j, feature i associated weights	w_{ji}, w^c_{ji}		w_{ji}, c_{ji}, w^c_{ji}	
Activation	$T_j = \dfrac{\lvert \mathbf{I} \wedge \mathbf{W_j} \rvert}{\alpha + \lvert \mathbf{W_j} \rvert}$		$\eta_{R_j}(\mathbf{I}) = \prod_{\mathrm{i}=1}^{\mathrm{M}} \eta_{\mathrm{ji}}(\mathbf{I_i})$ where $\eta_{ji}(I_i) = \begin{cases} \max\left(0, \dfrac{\gamma(I_i - w_{ji}) + 1}{\gamma(c_{ji} - w_{ji}) + 1}\right) & \text{if } I_i \le c_{ji} \\ \max\left(0, \dfrac{\gamma(1 - I_i - w^c_{ji}) + 1}{\gamma(1 - c_{ji} - w^c_{ji}) + 1}\right) & \text{if } I_i > c_{ji} \end{cases}$	
Learning laws (matching)		$\mathbf{W_J^{(new)}} = \beta(\mathbf{I} \wedge \mathbf{W_J^{(old)}}) + (1 - \beta)\mathbf{W_J^{(old)}}$		
	–		$\mathbf{C_J^{(new)}} = \beta^c \mathbf{I} + (1 - \beta^c)\mathbf{C_J^{(old)}}$	
Learning laws (error minimization)	–			$\mathbf{p(k)} = \mathbf{p(k-1)} - \epsilon(\mathbf{y} - \mathbf{d}) \dfrac{\partial \mathbf{y}}{\partial \mathbf{p}}\Big\vert_{\mathbf{k}-1}$ $\mathbf{p} = \mathbf{W}, \mathbf{C}$

error indices. Then, smaller network complexity was obtained for the same magnitude of error indices achieved by PROBART, which was trained on a much larger number of patterns.

The results achieved on these simple problems have been validated in other application-oriented works. In Gómez et al. (1998b) a highly nonlinear process, the penicillin fermentation, was simulated and FasArt was satisfactorily tested for biomass identification using 6–8 input variables in realistic conditions (noise and low sampling). In Araúzo, Gómez, Cano, Coronado, López & Collados (1999) and Gómez, Araúzo, Cano, Dimitriadis, López and López (1999) these results have been validated with data obtained from a penicillin pilot plant.

4. Conclusions

Neuro-fuzzy systems are a feasible solution to the problem of function identification from numeric data. As fuzzy logic systems, the performance of such systems can be described with a fuzzy rule base and a fuzzy inferencing engine. Furthermore, as neural networks, they provide learning capabilities for the automatic generation of these models. FasArt and FasBack are two architectures that maintain this dual interpretation: fuzzy logic systems and neural networks, due to the analogies between activation and membership functions, as well as neural connection and fuzzy rule.

FasArt architecture maintains the general structure of ARTMAP, preserving its main features, and adding the concept of fuzzy category. This is achieved by the definition of membership function of a category as the activation function of the related neural unit. Thanks to this duality, mechanisms from both fields can be applied in the same model, such as neural learning or defuzzification. Moreover, theoretical results from fuzzy systems literature characterize FasArt as a universal applicator. Besides learning by matching, the introduction of learning guided by prediction error minimization in FasBack architecture allows a reduction in network complexity while maintaining performance indices similar to those of FasArt.

Learning from noisy patterns is an important problem in the identification task, since real data are often corrupted by noise due to sensor inaccuracy among other reasons. Architectures within ART theory are highly sensible to this fact, since the winner-takes-all mechanism yields category proliferation, without enhancing prediction error. PROBART architecture reduces this problem, but it makes important modifications in ARTMAP structure, thus losing some of its main features.

Table 7 summarizes in a comparative fashion, the main features of the two proposed architectures and those of Fuzzy ARTMAP and PROBART. The proposed architectures are equivalent during the test stage to a fuzzy system with fuzzification by single point, inference by product and defuzzification by average of fuzzy sets, while this duality cannot be found in Fuzzy ARTMAP or PROBART. This is achieved by the definition of a new activation/membership function, as mentioned above. For the newly introduced weights $\mathbf{C_J}$ learning rules are introduced, in addition to those preserved from Fuzzy ARTMAP original algorithm. In FasBack, additional learning rules are defined to fine-tune weights in order to minimize prediction error.

The inter-ART map of FasArt architecture stores relations between F_2^a and F_2^b units. As well as for Fuzzy ARTMAP (in most practical applications $\rho^{ab} = 1$), each neuron in F_2^a may be linked to just one neuron in F_2^b. On the contrary, PROBART's inter-ART map stores probabilistic relations, so that a unit in F_2^a may be linked to several units in F_2^b, with different probabilities. Due to this fact, PROBART categorization of the input space is totally determined by ρ_a, while in Fuzzy ARTMAP, and FasArt and FasBack architectures, an inter-ART reset mechanism allows performing supervised learning.

In addition, we carried out experimental work on a problem proposed by PROBART designers, proving FasArt and FasBack capable of learning from noisy data. Results have shown very good performance of FasArt and FasBack, although they have not been specifically designed to cope with this particular problem. It has been seen that both FasArt and FasBack can learn from noisy data without meaningful loss of precision or increase of network complexity, as compared with results achieved when learning from clean data.

Acknowledgements

This research work was partially financially supported by the BRE2-CT94-0976 'PSYCHO' and ESPRIT 22416 'MONNET' projects of the European Union. The authors would like to thank all researchers of the Neurotecno Institute for their contributions, and especially M.A. Araúzo Bravo. The authors would like to thank the anonymous reviewers for their interesting comments that helped to improve the final quality of the paper.

References

Abe, S., & Lan, M. -S. (1995). Fuzzy rule extraction directly from numerical data for function approximation. *IEEE Transactions on Systems, Man and Cybernetics*, *25* (1), 119–129.

Araúzo, M. J., Gómez, E., Cano, J. M., Coronado, J., López, M.J., & Collados, A. (1999). Control of the penicillin production with adaptive IMC using fuzzy neural networks. In *Proceedings of 14th World Congress of IFAC* (pp. 499–504). Beijing, vol. O.

Bickel, P., & Rosenblatt, M. (1973). On some global measures of the deviations of density functions estimators. *Annals of Statistics*, *1*, 1071–1095.

Bishop, C. (1994). Training with noise is equivalent to Tikhonov regularization. Technical Report NCRG/4290, Neural Computing Research Group, Aston Univesity.

Blume, M., & Esener, S. (1995). Optoelectronic fuzzy ARTMAP processor. *Optical Computing*, *10*, 213–215.

Buckley, J. (1992). Universal fuzzy controllers. *Automatica*, *28* (6), 1245–1248.

Cano. J. M., Dimitriadis, Y. A., Araúzo, M. J., Abajo, F., & Coronado, J. (1996b). A neuro-fuzzy architecture for automatic development of fuzzy controllers. *In Proceedings of Computational Engineering in Systems Applications* (pp. 1187–1192). Lille, vol. 1.

Cano, J. M., Dimitriadis, Y. A., Araúzo, M. J., Abajo, F., & Coronado, J. (1996c). Fuzzy Adaptive System ART-based: theory and application to identification of biochemical systems. In *Proceedings of Computational Engineering in Systems Applications* (pp. 918–923). Lille, vol. 1.

Cano, J. M., Dimitriadis, Y. A., Araúzo, M. J., & Coronado, J. (1996a). FasArt: A new neuro-fuzzy architecture for incremental learning in systems identification. In *Proceedings of 13th World Congress of IFAC* (pp. 133–138). San Francisco, vol. F.

Cano, J. M., Dimitriadis, Y. A., & Coronado, J. (1997). FasBack: Matching error based learning for automatic generation of fuzzy logic systems. In *Proceedings of the sixth IEEE International Conference on Fuzzy Systems* (pp. 1561–1566). Barcelona, vol. 3.

Carpenter, G. & Govindavajan, K. (1993). Evaluation of speaker normalization methods for vowel recognition using Fuzzy ARTMAP and k-NN. Technical Report CAS/CNS-93-013, Boston University, Center for Adaptive Systems and Department of Cognitive and Neural Systems.

Carpenter, G., & Grossberg, S. (1987). ART 2: Self-organization of stable category recognition codes for analog input patterns. *Applied Optics*, *26* (23), 419–4920.

Carpenter, G. & Grossberg, S. (1989). Self-organizing neural network architectures for real-time adaptive pattern recognition. In *An introduction to neural and electronic networks* (pp. 455–478). Springer Verlag.

Carpenter, G., & Ross, W. (1995). ART-EMAP: a neural network architecture for object recognition by evidence accumulation. *IEEE Transactions on Neural Networks*, *6* (4), 805–818.

Carpenter, G. & Tan, A., Extraction, fuzzy ARTMAP, and medical database. In *Proceedings of the World Congress on Neural Networks*. Portland, vol. 1.

Carpenter, G., & Tan, A. (1995). Rule extraction: from neural architecture to symbolic representation. *Connection Science*, *7*, 3–27.

Carpenter, G., Grossberg, S., & Iizuka, K. (1992a). Comparative performance measures of fuzzy ARTMAP learned vector quantization and back-propagation for handwritten character recognition. In *Proceedings of the International Joint Conference on Neural Networks* (pp. 794–799). Baltimore, vol. 1.

Carpenter, G., Grossberg, S., Markuzon, N., Reynolds, J., & Rosen, D. (1992b). Fuzzy ARTMAP: a neural network architecture for incremental supervised learning of analog multidimensional maps. *IEEE Transactions on Neural Networks*, *3* (4), 698–713.

Carpenter, G., Grossberg, S., & Reynolds, J. (1991e). ARTMAP: supervised real-time learning and classification of nonstationary data by a self-organizing neural network. *Neural Networks*, *4* (5), 565–588.

Carpenter, G., Grossberg, S., & Reynolds, J. (1995). A fuzzy ARTMAP non-parametric probability estimator for nonstationary pattern recognition problems. *IEEE Transactions on Neural Networks*, *6*, 1330–1336.

Carpenter, G., Grossberg, S., & Rosen, D. (1991a). ART 2-A: an adaptive resonance algorithm for rapid category learning and recognition. *Neural Networks*, *4*, 493–504.

Carpenter, G., Grossberg, S., & Rosen, D. (1991c). Fuzzy ART, an adaptive resonance algorithm for rapid, stable classification of analog patterns. In *Proceedings of the International Joint Conference on Neural Networks* (pp. 411–416). Seattle, vol. 2.

Carpenter, G., Grossberg, S., & Rosen, D. (1991d). Fuzzy ART: fast stable learning and categorization of analog patterns by an adaptive resonance system. *Neural Networks*, *4*, 759771.

Carpenter, G., Milenova, B., & Noeske, B. (1998). Distributed ARTMAP: a neural network for fast distributed supervised learning. *Neural Networks*, *11*, 793–813.

Castro, J. (1995). Fuzzy logic controllers are universal approximators. *IEEE Transactions on Systems, Man, and Cybernetics*, *25* (4), 629–635.

Castro, J., & Delgado, M. (1996). Fuzzy systems with defuzzification are universal approximators. *IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics*, *26* (1), 149–152.

Chen, T., & Chen, H. (1995). Approximation capability to functions of several variables, nonlinear functionals, and operators by radial basis neural networks. *IEEE Transactions on Neural Networks*, *6* (4), 904–910.

Custodio, J. J., Tascón, M., Merino, M., & Dimitriadis, Y. A. (1999). In *Proceedings of the International Conference on Artificial Neural Networks* (pp. 964–969). Edinburgh.

Firmin, C. & Hamad, D. (1994). Gaussian based neural networks applied to pattern classification and multivariant probability density estimation. In *Proceedings of the second IEEE International Conference on Fuzzy Systems*. San Francisco.

Georgiopoulos, M., Fernlund, H., Bebis, G., & Heileman, G. (1996). Order of search in Fuzzy ART and Fuzzy ARTMAP: effect of the choice parameter. *Neural Networks*, *9* (9), 1541–1559.

Georgiopoulos, M., Heileman, G., & Huang, J. (1992). The N-N-N conjuncture in ART 1. *Neural Networks*, *5*, 745–753.

Georgiopoulos, M., Huang, J., & Heileman, G. (1994). Properties of learning in ARTMAP. *Neural Networks*, *7* (3), 495–506.

Gómez, E., Araúzo, M. J., Cano, J. M., Dimitriadis, Y. A. López, J., & López, M. J. (1999). Control of the penicillin production using fuzzy neural networks. In *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics* (pp. 446–450). Tokyo, vol. 6.

Gómez, E., Cano, J. M., Araúzo, M. J., Dimitriadis, Y. A., & Coronado, J. (1998b). Adaptive IMC using fuzzy neural networks for the control of non-linear systems. In *Proceedings of Conference on Integration in Manufacturing* (pp. 792–801). Göteborg.

Gómez, E., Gago, J. A., Dimitriadis, Y. A., Cano, J. M., & Coronado, J. (1998aa). Experimental study of a novel neuro-fuzzy system for on-line handwritten UNIPEN digit recognition. *Pattern Recognition Letters*, *19* (3), 357–364.

Goodman, P., Kaburlasos, V., Eubert, D., Carpenter, G., & Grossberg, S. (1994). Fuzzy ARTMAP neural network compared to linear discriminant analysis prediction of the length of hospital stay in patients with pneumonia. In *Fuzzy logic technology and applications*. IEEE Press.

Huang, J., Georgiopoulos, M., & Heileman, G. (1995). Fuzzy ART properties. *Neural Networks*, *8* (2), 203–212.

Jang, J. -S. R. (1993). ANFIS: Adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man, and Cybernetics*, *23* (3), 665–685.

Jokinen, P. (1992). On the relation between radial basis function networks and fuzzy systems. In *Proceedings of the International Joint Conference on Neural Networks*. Baltimore.

Kosko, B. (1992). Fuzzy function approximation. In *Proceedings of the International Joint Conference on Neural Networks*. Baltimore.

Lee, C. (1990). Fuzzy logic in control systems: fuzzy logic controller. Part II. *IEEE Transactions on Systems, Man, and Cybernetics*, *20* (2), 419–435.

Lin, C., & Lin, C. (1996). Reinforcement learning for an ART-based fuzzy adaptive learning control network. *IEEE Transactions on Neural Networks*, *7* (3), 709–730.

Malkani, A., & Vassiliadis, A. (1995). Parallel implementation of the Fuzzy ARTMAP neural network paradigm on a hypercube. *Expert Systems*, *12* (1), 39–53.

Marriott, S., & Harrison, R. (1995). A modified Fuzzy ARTMAP architecture for the approximation of noisy mappings. *Neural Networks*, *8* (4), 619–641.

Mulgrew, B. (1996). Applying radial basis functions. *IEEE Signal Processing Magazine*, *13* (2), 50–65.

Osman, H., & Fahmy, M. (1994). Probabilistic winner-take-all algorithm for radial-basis-function neural classifiers. *Neural Computation*, *6*, 927–943.

Roy, A., Govil, S., & Miranda, R. (1995). An algorithm to generate radial basis functions (RBF)-like nets for classification problems. *Neural Networks*, *6* (2), 179–201.

Rubin, M. (1995). Application of Fuzzy ARTMAP and ART-EMAP to automatic target recognition using radar range profiles. *Neural Networks*, *8* (718), 1109–1116.

Sainz, G. I., Dimitriadis, Y. A., Cano, J. M., Gómez, E., & Parrado, E. (2000). ART based model set for pattern recognition: FasArt family. In H. Bunke, & A. Kandel (Eds.), *Neuro-fuzzy pattern recognition*. World Scientific Publ. Co.

Salzberg, S. (1991). A nearest hyperrectangle learning method. *Machine Learning*, *6*, 251–276.

Serrano Gotarredona, T., & Linares Barranco, B. (1996). A modified ART1 algorithm more suitable for VLSI implementations. *Neural Networks*, *9* (6), 1025–1044.

Simpson, P. (1993). Fuzzy min–max neural network for function approximation. In *Proceedings of the International Conference on Neural Networks*. San Francisco.

Srinivasa, N. (1997). Learning and generalization of noisy mapping using a modified PROBART neural network. *IEEE Transactions on Signal Processing*, *45* (10), 2533–2550.

Wang, L. (1994). *Adaptive fuzzy systems and control*, New Jersey: Prentice Hall.

Williamson, J. (1996). Gaussian ARTMAP: a neural network for fast incremental learning of noisy multidimensional maps. *Neural Networks*, *9* (5), 881–897.

Zadeh, L. (1965). Fuzzy sets. *Information and Control*, *8*, 338–353.

Zadeh, L. (1988). Fuzzy logic. *Computer*, *21* (4), 83–93.